July 25, 2018

AUTO DETECTION FOR RED SEAL AND IMPROVING PRINTING QUALITY FOR IT

HP INC

Auto Detection for Red Seal and Improving Printing Quality for it

A method for automatically optimizing the printing quality of red seal in documents is disclosed. This method involves the red circle seal detection in the documents to be printed.

Usually the red circle seal in government or big enterprise official documents can represent the authority, so the customers require the red circle seal should be very vivid in the printout. Some printers' SW driver can allow customers to manually select a specific printing mode for document with red circle seal before printing, this is not user friendly and it cannot improve the PQ (printing quality) for the red seal precisely. Now this disclosure aims to provide a built-in solution to auto detect the red circle seal and precisely improve the red circle seal area's PQ. This solution is to make the original document as a picture first, then convert the picture from RGB(red/green/blue) space to HSV (Hue, Saturation, Value) space, in HSV space the thick vivid red components could be kept but the other parts of the picture could be filtered out by setting thresholds, then for this remainder consisting of thick vivid red elements, it's able to detect whether there is circle shape or not by using the Hough Transform in OpenCV (Open Source Computer Vision Library). Furthermore, the coordinate (x, y) of the center of the circle and the radius R of the circle could be known from Hough Transform circle detection result, so these data could be sent to low level FW, FW can choose specific color table to print for the specific area where the red circle seal resides.

This solution could be embedded in printer SW.

The flow chart show details.

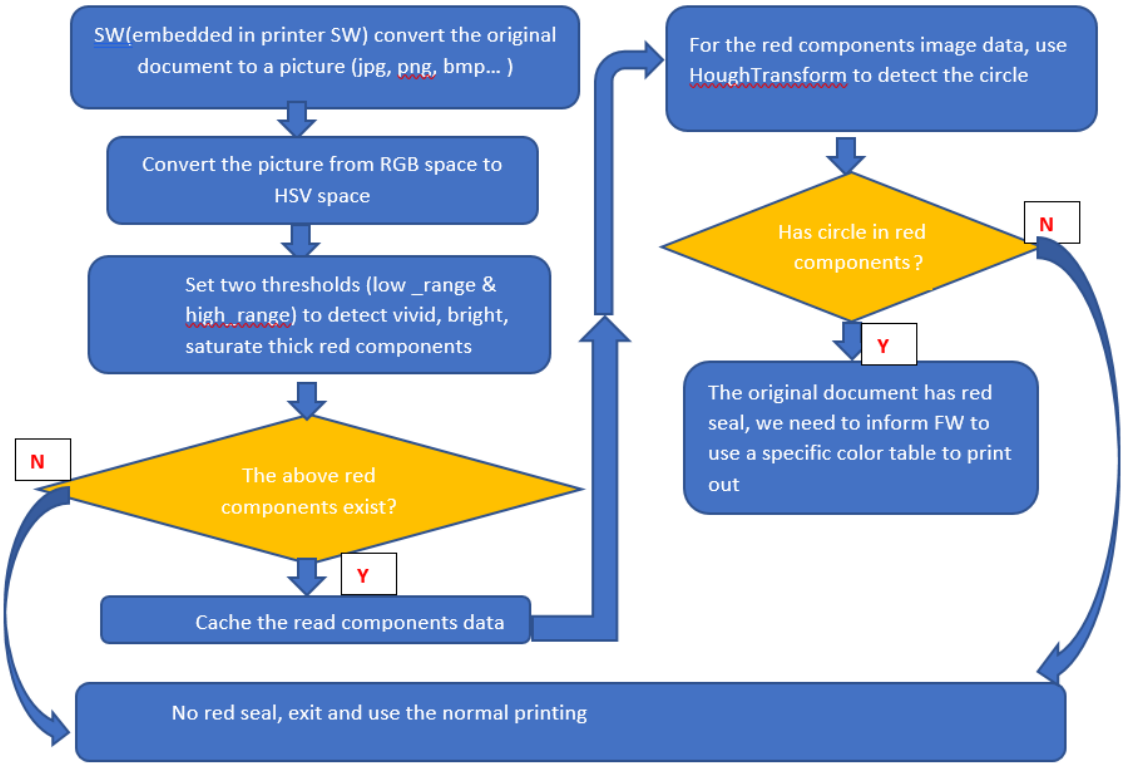

Figure-1

A demo has been shown below (The demo is using open CV library and python 3.6).

Original document:

Figure-2

HSV space filter:

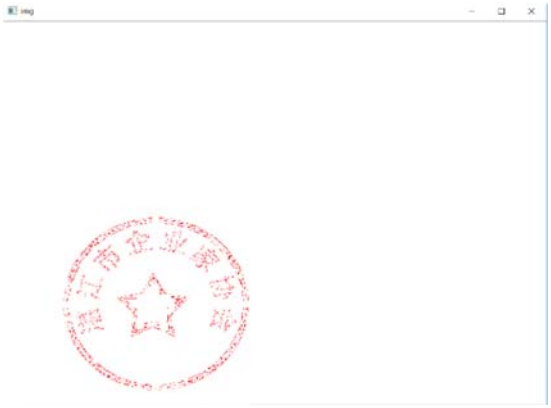

Figure-3

Hough Transform circle detection

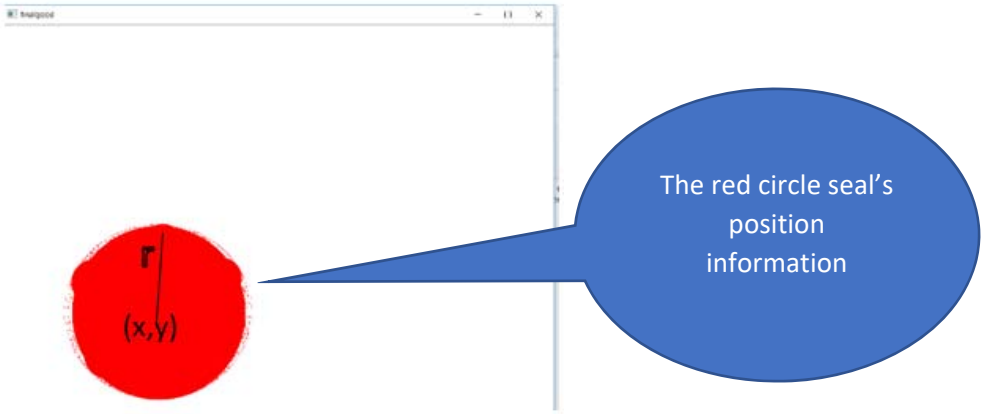

Figure-4

Python code piece,

circles= cv2.HoughCircles(gray,cv2.HOUGH_GRADIENT,1,100,param1=100,param2=30,minRadius=5,maxRadius=300)

x=int(circle[0]) # coordinate x of the center of the circle

y=int(circle[1]) # coordinate y of the center of the circle

r=int(circle[2]) # this is radius

Disclosed by Jian-Qun Zhang, Jing Zhao and Lin-Lin Hua, HP Inc.